Can the operator of a drone be located by following the drone's path?

Eliyahu Mashhadi, Yossi Oren, and Gera Weiss

Ben Gurion University of The Negev, Israel

Abstract. Small commercial Unmanned Aerial Systems (UASs), called drones in common language, pose significant security risks due to their agility, high availability and low price. There is, therefor, a growing need to develop methods for detection, localization and mitigation of malicious and other harmful operation of these drones. This paper presents our work towards autonomously localizing drone operators based only on following their path in the sky. We use a realistic simulation environment and collect the path of the drone when flown from different points of view. A deep neural network was trained to be able to predict the location of drone operators, given the path of the drones. The model is able to achieve prediction of the location of the location of the operator with 73% accuracy.

Keywords: Drone · UAS · Surveillance · Security · Deep Learning · Deep Neural Network.

1 Introduction

The massive use of drones for civilian and military applications raises many concerns for airports and other organizations [4]. In December 2018, for example, drones infamously caused the shutdown of the Gatwick airport in the United Kingdom. This also happened in Germany, where a drone caused the suspension of flights in Frankfurt. As the threats that drones incur include also surveillance and active attacks, defense agencies are looking for ways to mitigate the risks by locating and tracking operators of drones [3].

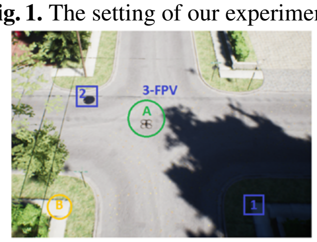
A number of different sensor types are available for the detection and localisation of drones and their operators. The most common sensor types studied by the research community used commercially are: Radio Frequency (RF) [6, 10], Electro-Optical (EO), acoustic and radar. All the approaches that we are aware of for locating operators, not just the drones, use RF sensors. There are automatic and semi-automatic methods for locating the operators based on the radio communication between the drone and its operator. There are a number of problems with this approach. Firstly, such methods are usually tailored to a specific brand of drones. Furthermore, the radio signal can only be recorded near the drone. Finally, there are ways for malicious drone designers to apply cryptography and electronic warfare techniques to make localization by analysis of radio signals very difficult.

In this work we propose a novel method for the localisation of UAS operators using only the path of the drone in the sky. The approach is based on the observation that the behaviour of a drone in the air is visibly different depending on where the pilot is. An experienced external viewer can usually tell if the pilots uses First-Person-View (FPV) machinery or if they look at the drone from east or if they look at it from a distance. We assume that the defenders are capable of tracking the path of the drone in the sky, and show that this information is enough to gain valuable information on the location of the operator. While the path can be measured from a relatively large distance [1], it contains information because the operators usually react to environmental conditions such as sun dazzle, obstructions, etc. Our experiments show that the reactions of the operators to these conditions gives away enough information for obtaining substantial information about the location of the operator by analyzing the path of the drone in the sky. Note that we are not necessarily aiming for full localization in all setting, even the ability of distinguish between three different operators, looking from three different points of view, carrying the same known task (which is what we demonstrate in this paper) can be useful for defenders. For example, the defenders of an airport cad use such knowledge to block the line of sight of the pilot of an infiltrating drone. To the best of our knowledge, we are the first to provide a data-set of flight-paths labeled with the point-of-view of the operator and to train neural networks on such data.

2 Methodology

To allow for a controlled environment, we conducted all our experiments with a flight simulator that provides a realistic flight experience for the operator that includes sun gazes, obstructions, and other visual effects that produce the reactions of the operators that allow us to identify their location. Specifically, we used AirSim (Aerial Informatics and Robotics Simulation), which is an open-source, cross platform simulator for drones, ground vehicles such as cars and various other objects, built on Epic Games' Unreal Engine 4 [5]. AirSim provides more than 10 kilometers of roads with many city blocks. We used it via its API that allowed us to retrieve data and control drones in a safe environment. AirSim supports hardware-in-the-loop with driving wheels and flight controllers physically and visually realistic simulations. This allowed us to provide drone pilots with a real remote control and a simulation of the full piloting experience, including the artifacts that cause pilots to perform maneuvers that unintentionally disclose their position to the defenders that watch the path of the drone.

Fig. 1. The setting of our experiments.



As shown in Figure 1, we collected the path of the drone when flown from three different viewpoints. Two points, marked with 1 and 2, on two opposite sides of the intersection and a third point, marked by 3, from First Person View (FPV) where the operator gets the perspective of a real pilot that seats aboard the drone. In all the experiments the pilots were instructed to fly the drone from point A, in the middle of the intersection, to point B, at the bottom left.

Fig. 2. A log of a flight produced by AirSim.

1	TimeStamp	POS_X	POS_Y	POS_Z	Q_W	Q_X	Q_Y	Q_Z
2	1557313139904	0.020689	0.025806	-1.014527	0.995622	0.001442	-0.001119	0.093454
3	1557313140027	0.024383	0.030536	-1.035369	0.991322	0.001469	-0.001079	0.131444
4	1557313140150	0.028316	0.035702	-1.044194	0.985579	0.001503	-0.001028	0.169208
5	1557313140276	0.032581	0.041509	-1.054980	0.979984	0.001545	-0.000964	0.199066
6	1557313140411	0.037385	0.048357	-1.077708	0.975740	0.001586	-0.000898	0.218925
7	1557313140537	0.042059	0.055367	-1.114959	0.973107	0.001614	-0.000851	0.230346

The results of the experiments were files, such as the one presented in Figure 2, containing the log of the flight produced by AirSim. This simulates the data that we expect that the defenders can collect. It contains the full path information including the position, the orientation, and the picture of the drone in each time step. As elaborated below, we did not always use all this information with full accuracy, because it is not necessarily available.

Fig. 3. A comma separated file ready to be used for machine learning..

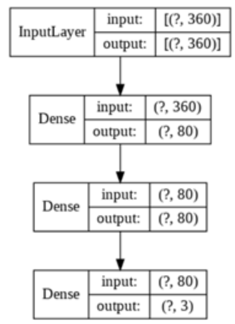
```
!head -n7 {TRAIN_DATASET_FILE_PATH}

57,360,FPV,north_west_tree,south_east_sewer
1.8e-06,2.5333333333333334e-06,-0.004928333333333334,3.626666666666667e-05,4.473333333333334e-05,-0.00536706666666
1.0000000000000001e-07,1.3333333333333334e-07,-0.004885833333333334,1.8333333333333335e-06,2.3666666666666667e-06,-
2.333333333333333e-06,1.0333333333333333e-06,-0.004908133333333334,1.3366666666666667e-05,5.666666666666667e-06,-0
6.666666666666667e-08,-2.6666666666666667e-07,-0.004909966666666667,1.8333333333333335e-06,-5.666666666666667e-06,-
2.3333333333333333e-07,3.3333333333333335e-07,-0.005110999999999995,4.666666666666666e-06,6e-06,-0.00664216666666
0.000738033333333334,0.04812483333333333,-0.01976413333333333,0.002752499999999997,0.05618766666666664,-0.02191
```

We then parsed these text files and translated them to the format shown in Figure 3 that is more amenable for efficient machine learning tasks. The data-set that we have created is publicly available and is considered one of the contributions of this paper. The data-set contains 81 flights, 27 from each operator location (A, B, or C). Each flight is represented by a file with 360 features consisting of 120 (X,Y,Z) triplets, each representing the position of the drone at a specific time along the flight. The location of the drone was captured in 8 Hertz, i.e., we recorded an (X,Y,Z) triplet every 125 milliseconds.

3 Results

In this section we report on the main results we obtained with our experiments.

Fig. 4. A dense neural network we used for identification of the location of the drone's operator.



3.1 The path of the drone gives away information on the location of the pilot

We used the data-set described in Section 2 to train neural networks with different parameters and typologies, as shown in Figure 8. The topology that yielded the best results is built of two dense layers as shown in Figure 4. It allowed us to demonstrate that it is possible to infer significant information about the location of the operator form by analyzing the path of the drone.

Fig. 5. The variations of the neural networks produced by our script.

batch size.	number of neurons	epochs	activation function	Accuracy
10	80	13	relu	73.99
20	80	13	sigmoid	73.99
20	80	10	relu	73.64
10	20	10	elu	73.64
20	80	13	relu	73.28
10	20	8	relu	73.28
20	20	13	sigmoid	72.92
10	80	10	relu	72.92
10	20	13	elu	72.57
10	80	8	sigmoid	72.57
⋮	⋮	⋮	⋮	⋮

We repeated the training and quality measurement process many times with an automatic script that created a variety of similar models by varying the parameters of the model shown above. We chose the variation of the model that produced the best results, and tested its accuracy with respect to records in the data set that were not used for training. This model was able to guess the viewpoint of the operator with 73% accuracy.

3.2 The orientation of the drone is not needed

Beyond location, the defender that observes the drone can also measure its Euler angles. Because such measurements may require more expensive equipment mounted closer to the drone, we ran experiments to measure how much this information can contribute to the accuracy of identification of the pilot's point-of-view.

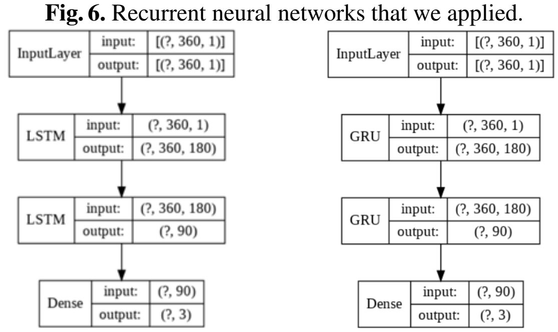
To this end, we extended our data-set with information about the orientation of the drone along its flight. When trained and tested with both location and orientation data, our neural networks achieved accuracy of 74%, which is a one percent improvement over the accuracy we obtained with location information only. When trained with orientation data only, the performance degraded a little to 71% precision. Our conclusion is that it seems that there is no need for measuring the orientation of the drone, if this entails costs and limitations.

Our explanation to the fact that the orientation information did not contribute much to the accuracy of the inference is that the location and the orientation variables are coupled. Specifically, the speed of the drone in each direction is a direct function of the thrust of the rotors and the Euler angle that corresponds to that direction. Thus, the location of the drone can be inferred within some error margins by integrating its rotations on all axes. Evidently, the neural network that we have designed was able to take advantage of these relations when we asked it to use only position or only rotation information.

3.3 Recurrent networks are not better for the task

Since our motivation was to identify temporal patterns in the data, we thought that it may be possible to improve the accuracy of the network in performing the required task by applying a recurrent neural network (RNN). Such networks have a temporal dimension so they can handle time and sequences.

Fig. 6. Recurrent neural networks that we applied.



We tried the recurrent topologies depicted in Figure 6. As shown in Figure 7, these networks yielded only 55% accuracy. We do not know how to explain this performance degradation.

3.4 The effect of measurement disturbances, measurement accuracy and and sampling rate

While we ideally want to measure the a time-varying position of the drone so we can accurately reconstruction of the signal from collected discrete data points, the sampling

Fig. 7. Recurrent neural networks that we applied.

```
LSTM:
average_result:55.0000059604645, number_of_neurons:130, activation_func:<function softplus at 0x7f7ad3034c80>,
average_result:52.8571441769599, number_of_neurons:150, activation_func:<function sigmoid at 0x7f7ad304d950>,
average_result:50.00001192092896, number_of_neurons:130, activation_func:<function sigmoid at 0x7f7ad304d950>,
average_result:48.571429550647736, number_of_neurons:180, activation_func:<function softplus at 0x7f7ad3034c80>
average_result:48.571429550647736, number_of_neurons:100, activation_func:<function softplus at 0x7f7ad3034c80>
average_result:47.85714358091354, number_of_neurons:80, activation_func:<function softplus at 0x7f7ad3034c80>,
```

```
GRU:
average_result:57.85714417695999, number_of_neurons:130, activation_func:<function sigmoid at 0x7f7ad304d
average_result:55.71428745985031, number_of_neurons:180, activation_func:<function sigmoid at 0x7f7ad304d
average_result:53.57142984867096, number_of_neurons:150, activation_func:<function softplus at 0x7f7ad303
average_result:50.00000089406967, number_of_neurons:180, activation_func:<function softplus at 0x7f7ad303
average_result:49.285715222358704, number_of_neurons:80, activation_func:<function softplus at 0x7f7ad303
average_result:48.571429550647736, number_of_neurons:100, activation_func:<function sigmoid at 0x7f7ad304
average_result:47.1428582072258, number_of_neurons:80, activation_func:<function sigmoid at 0x7f7ad304d95
average_result:47.142857909202576, number_of_neurons:150, activation_func:<function sigmoid at 0x7f7ad304
```

speed and precision of the measurement instruments can directly affect the ability to reconstruct the signal [7]. Ideally, the measurement infrastructure captures the signal continuously with perfect accuracy (precision and trueness). But in reality, many devices sample signals discretely. And they are affected by noise. systematic noise affects trueness, while random noise compromises precision. Clearly, the more information about the signal we can capture with the data points, the better accuracy. Where necessary, the amount of signal data can be increased by collecting more samples per unit of time, and by improving the signal-to-noise ratio of each sample.

Figure 8 shows the trade-off between sampling rate and precision. The table shows that, as expected, reducing the sampling frequency reduces the accuracy rather dramatically. This shows that the identification of the position of the operator of the drone relies on relatively high frequency properties of the signal, i.e., on variation of the path that can only be detected when the position of the drone is sampled at high frequency.

Fig. 8. The effect of the sampling rate on accuracy.

Rate Hz.	**Time diff.** seconds	**Accuracy**
8	0.125	73.57
4	0.25	67.5
3	0.375	60.35
2	0.5	56.07
4/3	0.75	47.28
8/7	0.875	44.64
1	1	40.71

Figure 9 shows the trade-off between the sampling precision and the accuracy of the estimation. This data shows that our ability to estimate where the operator of a drone is does not drop very dramatically when the location of the drone is measured with lower precision. This indicates that the maneuvers that the network bases its estimation upon are relatively wide, i.e., we see that the network is able to detect the differences even with a precision level of one decimeter.

Figure 10 shows the effect of sampling disturbances on the accuracy of the estimation. This data shows that event with noise that add up to 5 meters to the measurement, the neural network is able to maintain high estimation accuracy. This data indicates that the network is capable to ignore the distur

Fig. 9.

Sampling precision meters	**Estimation Accuracy**
10^{-4}	73.57
10^{-3}	72.85
10^{-2}	72.85
10^{-1}	68.21
1	37.85

Fig. 10.

Sampling Disturbance $Uniform[0,x]$ meters	**Estimation Accuracy**
0	73.57
1	67.14
5	62.14
10	48.57
15	46.43

The effect of the sampling precision on estimation accuracy.

4 Related Work

The usual way for locating drone operators is via RF techniques. Locating drone signals can be a challenge due to the the amount of other WiFi, Bluetooth and IoT signals in the air. Drone operation radio signal have short duration, their frequency usually hops over most of the band and they have relatively low power. To effectively collect these signals, network-enabled sensors must be distributed around the flight area so the defenders can detect and locate the needed signals. For successful pinpointing of the operator, the signals should be isolated in frequency and time. After detecting the RC, the geolocation system must triangulate the signal using data it collects from the sensors. Since broad scanning of all the traffic is expensive due to sensor allocation and computational complexity, our work may complement RF based system by narrowing the search to more probable areas based on the drone path, which is easier to follow.

Another way that our work can complement RF based technique is by the observation that there is a strong association between the maneuvers of the drone and the command patterns sent via RF. This may allow to solve a crucial issue with RF based techniques that have trouble identifying the signal related to a specific drone in an urban environment where many similar signals (possible, even, from other drones of the same brand). We can train our neural networks to identify command patterns of the signal transmitted from the operator when the drone is turning, rotating, accelerating, and decelerating and use it to connect a signal to a specific drone in the air.

Lastly, RF based techniques can only detect the antenna from which the signal is sent. This may allow to intercept that antenna, but malicious operators can easily redirect their signal to another antenna without interrupting their mission. Our technique allows to get direct information about the viewpoint of the operators which allows more effective interception. Even identifying that the operator uses the FPV viewpoint can be useful, because the defenders can distract this view by clouding the area of the drone.

In the technical level, our work is also related to driver identification [9, 8, 2]. Models of driving behaviors is an active field of research since the 1950s. Because driving consists complex actions and reactions, different driver express different driving behaviors in different traffic situations. These differences can be detected by observing how drivers use the pedals, the way they use the steering wheel, how they keep their eyes on the road, how much distance they keep from the other cars, and many other factors. There is much work on using neural networks for translating sensory data that is easily collected while driving to an educated guess of who is currently driving the car. This work is related to ours in that it also tries to use machine learning for inference of hidden information from human behaviour. It is interesting to note that while recurrent networks are the state of the art in the domain of driver identification, we obtained better performance with dense networks.

5 Conclusions and Future Work

Our initial results indicate that observing the path of a drone can indeed serve to identify the location of the drone's operator. It would be interesting to explore what additional data can be extracted from this information. Possible insights would include the technical experience level of the drone operator, where was the drone operator trained in flying, and possibly even the precise identity of the operator. Another direction would be in improving the machine learning pipeline. It would be interesting to compare different deep learning architectures, especially those tailored for the treatment of time-series data. The data-set used for training and evaluating our models is naturally smaller than machine-generated corpora used for other tasks such as malware classification. As such, it would be interesting to look for a feature set which can be used as input to a classical machine learning algorithm such as KNN or SVM, which traditionally requires less data than deep learning models.

References

1. J. Drozdowicz, M. Wielgo, P. Samczynski, K. Kulpa, J. Krzonkalla, M. Mordzonek, M. Bryl, and Z. Jakielaszek. 35 ghz fmcw drone detection system. In *IRS16*.
2. S. Ezzini, I. Berrada, and M. Ghogho. Who is behind the wheel? driver identification and fingerprinting. *J. Big Data*, 5:9, 2018.
3. M. García, A. Viguria, G. Heredia, and A. Ollero. Minimal-time trajectories for interception of malicious drones in constrained environments. In *ICCVS19*.
4. J. Russo, D. Woods, J. S. Shaffer, and B. A. Jackson. Countering threats to correctional institution security: Identifying innovation needs to address current and emerging concerns. RAND Corp. 2019.
5. S. Shah, D. Dey, C. Lovett, and A. Kapoor. Airsim: High-fidelity visual and physical simulation for autonomous vehicles. In *Field and service robotics*, pages 621–635. Springer, 2018.
6. D. Solomitckii, M. Gapeyenko, V. Semkin, S. Andreev, and Y. Koucheryavy. Technologies for efficient amateur drone detection in 5g millimeter-wave cellular infrastructure. *IEEE Communications Magazine*, 56(1):43–50, 2018.
7. C.-C. Wang and M. Tomizuka. Iterative learning control of mechanical systems with slow sampling rate position measurements. In *ASME08*.
8. J. Zhang, Z. Wu, F. Li, J. Luo, T. Ren, S. Hu, W. Li, and W. Li. Attention-based convolutional and recurrent neural networks for driving behavior recognition using smartphone sensor data. *IEEE Access*, 7:148031–148046, 2019.
9. J. Zhang, Z. Wu, F. Li, C. Xie, T. Ren, J. Chen, and L. Liu. A deep learning framework for driving behavior identification on in-vehicle CAN-BUS sensor data. *Sensors*, 19(6):1356, 2019.
10. J. Zhen. Localization of unmanned aerial vehicle operators based on reconnaissance plane with multiple array sensors. *IEEE Access*, 7:105354–105362, 2019.